CERTIFICATION OF TRANSLATION

I, ***Myung-kyo Kim***, an employee of Y.P. LEE, MOCK & PARTNERS of Koryo Building, 1575-1 Seocho-dong, Seocho-gu, Seoul, Republic of Korea 137-875, hereby declare under penalty of perjury that I understand the Korean language and the English language; that I am fully capable of translating from Korean to English and vice versa; and that, to the best of my knowledge and belief, the statement in the English language in the attached translation of ***Korean Patent Application 10-2002-0054945*** consisting of 54 pages, have the same meanings as the statements in the Korean language in the original document, a copy of which I have examined.

Signed this 19th day of May 2010

Myung Kyo Kim

A B S T R A C T

[Abstract of the Disclosure]

Provided are a multimedia data recording apparatus, a multimedia data reproducing apparatus, and multimedia data storage. The multimedia data storage medium contains multimedia data and includes a first layer in which multimedia data is stored; and a second layer in which, when the multimedia data is divided into a record unit and a reproduction unit, information regarding attributes of the record unit and relationship between the record unit and the reproduction unit are described with a markup language using elements and attributes. Accordingly, it is possible to add new multimedia data even if the standard has already been determined, and it is easy to implement extension of a future standard even if a record unit or a unit or reproduction is newly prescribed.

[Representative Drawing]

FIG. 15

S P E C I F I C A T I O N

[Title of the Invention]

APPARATUS FOR RECORDING OR REPRODUCING MULTIMEDIA DATA USING HIERARCHICAL INFORMATION STRUCTURE AND INFORMATION STORAGE MEDIUM THEREOF

[Brief Description of the Drawings]

FIGS. 1 through 13 illustrate the prior art; and

FIGS. 14 through 19 illustrate the present invention.

[Detailed Description of the Invention]

[Object of the Invention]

[Technical Field of the Invention and Related Art prior to the Invention]

The present invention relates to recording and reproducing multimedia data, and more particularly, to apparatuses for recording and reproducing multimedia data using a hierarchical information structure and information storage medium thereof.

A multimedia data recording/reproducing apparatus requires additional information, such as the attributes of multimedia information or the sequence of data reproduction, to record multimedia information on or reproduce multimedia information from an information storage medium. In general, in order to record the additional information, additional information tables are formed at a particular position or in a particular file of the existing data storage medium. Examples of data storage medium include compact discs (CDs), video CDs, and digital versatile discs (DVDs). In the additional information table, information is described by the location and length of a data field. Thus, a new information table needs to be created whenever a new type of multimedia information medium is designed. In particular, navigation information, for selecting a reproduction unit or determining the reproduction sequence, is recorded in table formats on a DVD. However, such navigation information is mixed with

information that defines reproduction units, thereby making it difficult to carry out navigation.

To solve this problem, the present invention suggests that such additional information be hierarchically recorded and a markup language be used to implement each layer for extensibility. The markup language, which describes record units and reproduction units, is called a media description language.

The data structure of a conventional DVD will now be described. Referring to FIG. 2, a DVD includes a video manager (VMG) area and a plurality of video title set (VTS) areas. Control information and title selection information are stored in the VMG area, while title information which is a reproduction unit is stored in the plurality of VTS areas. The VMG area consists of 2 or 3 files and each VTS area consists of 3 to 12 files.

The VMG area is illustrated in FIG. 3 in detail. The VMG area includes a video manager information (VMGI) area for storing additional information regarding the VMG area, a video object set (VOBS) area for storing video object information of a menu, and a VMGI backup area. Each of the above areas comprises a single file. The VOBS area may not be included in the VMG area, but the other two areas are required.

Title information and VOBS are stored in the VTS area. A plurality of titles may be recorded in the VTS area. A VTS area is illustrated in detail in FIG. 4.

Referring to FIG. 4, video title set information (VTSI); VOBS for menu which is a video object set for the VTS menu; VOBS for title, which is a video object set for titles in a VTS; and VTSI backup data are recorded in VTS area #n. However, the VOBS for title may not be included in VTS area #n. Each VOBS is divided into a video object (VOB) on cells, which are record units. Each VOB consists of a plurality of cells. In the present invention, a cell is determined to be the lowest-ranked unit of data.

A reproduction unit is represented by a hierarchical structure in which a title is present at its top level. The title includes one program chain (PGC) or a plurality of PGCS linked to one another. Here, a first PGC is called an entry PGC. FIG. 5 illustrates a reproduction unit *one_sequential_PGC_title* consisting of only a single PGC

(an entry PGC). FIG. 6 illustrates a reproduction unit in which a title is linked to a plurality of PGCs. After reproduction of a PGC, another PGC is reproduced by selecting from a group of several PGCs. In this case, a selection command may be stored in program chain information (PGCI). Controlling the sequence of PGC reproduction is called navigation. FIG. 7 illustrates the structure of a PGC. A PGC is stored in an information structure described as PGCI format. A PGCI includes a pre-command in which navigation commands are stored, a post-command, and a plurality of program information units. The pre-command is carried out before reproduction of a related PGC and the post-command is carried out after reproduction of the PGC. Each program information unit consists of a plurality of cell information units, each cell linked to a cell in a VOB, which is a record unit. Each cell included in each reproduction unit has a cell command that is carried out after reproduction of the cell. Therefore, the PGCI represents the hierarchical reproducing structure of a PGC, i.e., a reproduction unit, in which the lowest-ranked reproduction unit cell is linked to the lowest-ranked record unit cell.

FIG. 8 illustrates the case of branching a new PGC using command navigation information during or after reproduction of a PGC. In other words, it is possible to easily control the sequence of PGC reproduction using navigation commands such as LinkPrevPGC, LinkTopPGC, LinkNextPGC, LinkTailPGC, LinkGoUpPGC, and LinkPGCN. That is, a PGC has reproduction units and also navigation information.

A program in a PGC is referenced by a link called a part of title (PTT).

In the case of a DVD, the above information is stored in a binary table format, that is, information is recorded in table formats where information is recorded within a predetermined bit length on a particular position of the table.

FIG. 9 illustrates the data structure of a TT_SRPT information table, which is title information in VMGI. Referring to FIG. 9, the leading two bytes of the TT_SRPT information indicates a total number of titles *n*. The next two bytes of the TT_SRPT information is reserved for extension information defined in a future standard. The other bytes following the two bytes represent TT_SRP information that individually

describes titles. In particular, a VTN number, designated by a related title, and a title number in a related VTS are recorded with predetermined bit lengths on certain positions of VTSN and VTS_TTN, respectively.

FIG. 10 illustrates the data structure of VTS_PTT_SRPT information in VTSI. Referring to FIG. 10, the VTS_PTT_SRPT information includes TTU_SRPs as many as the titles of a related VTS. The respective TTU_SRPs contains information for designating one of the PTT_SRPs following the TTU_SRPs. Therefore, a PTT_SRP between a PTT_SRP, which is designated by a TTU_SRP, and a PTT_SRP, which is designated by the next TTU_SRP, forms a title. That is, a plurality of Part_of_Title Search Pointers *PTT-SRPs* are present in each title, resulting in division of a title into several PTTs. PTT may be viewed as a chapter, for easier understanding. FIG. 11 illustrates the contents of PTT_SRP, designating a PGC and a program in the PGC. In conclusion, a title is divided into several PTTs and each PTT is linked to a program in a PGC.

FIG. 12 illustrates the data structure of a PGCIT *VTS_PGCIT*, representing a PGCI table in VTSI. Referring to FIG. 12, the total number of programs and cells that belong to a PGC are stored in VTS_PGCITI. Also, the VTS_PGCIT stores VTS_PGCIs as many as a plurality of VTS_PGCs, which belong to a VTS. FIG. 13 illustrates the detailed data structure of VTS_PGCI. In VTS_PGCI, various information is recorded within particular bit lengths at particular positions in table formats, including PGC_CMDT which describes a pre command, a post command, a cell command, and so on; PGC_PGMAP which indicates program start cells as many as programs; and C_POSIT which is information for linkage of respective cells to respective record units.

As described above, in the case of a conventional DVD which is a multimedia data storage medium, video object data and data regarding reproduction units and navigation are recorded as titles and PGCs. However, the additional data in PGCs is stored in table format within particular lengths at a certain position. Commands for navigation are also stored in a limited space, together with additional data which define reproduction units. Therefore, the advent of a new reproduction unit results in a

change of the table location, thus making it difficult to implement extension for the new reproduction. To solve this problem, reserved spaces are formed in a plurality of regions, but this solution still limits any extension for the a future. In the worst-case scenario, a table structure may be redefined. In fact, existing multimedia data storage media such as CDs, VCDs, MDs, and DVDs have table structures of their own.

Meanwhile, advancements in Internet technologies trigger development of personal computers (PCs) capable of processing multimedia. For instance, video object data or audio data may be inserted into a document with HyperText Markup Language (HTML) or may be reproduced during reproduction of HTML. In this case, navigation data is described with a script language or the like. Therefore, it is preferable that navigation data is described separate from reproduction data. If there are two types of navigation data, e.g., one controlled using script language and another one described in table format, it is complicated to control both types of navigation data.

[Technical Goal of the Invention]

In this disclosure, the present invention is described with respect to a data storage medium and a data reproduction apparatus. Assuming that the data reproduction apparatus operates as specified in the present invention, a recording apparatus records information, according to the data structure of which is specified in the present invention, and multimedia data on a storage medium. Here, storing operations are understood as identical to recording operations.

Multimedia data and additional data are recorded in a storage medium according to the present invention. The present invention suggests separate recording of two different types of additional data: additional information regarding record units, attributes, and reproduction units of the multimedia data; and navigation information regarding selection of a reproduction unit and reproduction sequence. The additional information regarding a record unit, attributes, or a reproduction unit, with respect to multimedia data, is described using a markup language. Accordingly, implementation supporting the extension of a future standard is possible even when adding a new type of

multimedia data or prescribing a new type of record or reproduction unit, regardless of the standard adopted. Alternatively, the additional information may be stored in binary table format.

Also, both or one of a markup language and script language is used to describe navigation data, which represents selection of a reproduction unit or reproduction sequence. A markup language is also used to describe presentation data which represents a menu screen for selection of a reproduction unit and a screen layout for data reproduction, thereby enabling a menu structure and navigation with a high degree of flexibility.

[Structure and Operation of the Invention]

According to an aspect of the present invention, there is provided a multimedia data storage medium in which multimedia data is stored, the medium including a first layer in which multimedia data, such as video object images, still images, voice, graphics, and texts, is stored; and a second layer in which, when the multimedia data is divided into a record unit and a reproduction unit, information regarding attributes of the record unit and relationship between the record unit and the reproduction unit are described with a markup language using elements and attributes.

Navigation data, which is used to control selection of the reproduction unit and reproduction sequence, may be recorded on a third layer using a markup language or a script language, in addition to the information recorded on the first and second layers.

Preferred embodiments of the present invention will be described with respect to a storage medium on which a video object title is recorded, being supported by a plurality of voice types and subtitles. Referring to FIG. 14, file1.mpg, which is the first half of data representing the video object title, and file2.mpg, which is the other half, are recorded in a first layer. A title may be divided into a plurality of files given the size limit of a chapter unit or a file. Video object data is compressively encoded to reduce the data amount thereof. MPEG, which is one of the most popular motion picture compression methods, supports a variable bit rate (VBR) encoding method in which the

bit rate per hour varies according to the amount of video information. Time search determining the starting point of data reproduction with consideration to a given time requires table-type information for mapping the time and position of data. This information is used to detect the position of desired data since a predetermined time after the start of data reproduction. In general, the table-type information contains information regarding data positions measured at every predetermined point of time. For instance, the table-type information may be time map information that represents temporal position linking information indicating the data positions measured with respect to a file beginning every 10 seconds. However, because the amount of time map information is significant, the information is recorded in binary table format, rather than using markup language, and stored in the first layer. FIG. 14 illustrates time map information *file1timemap.dat* and *file2timemap.dat*, regarding video object data file1.mpg and file2.mpg, respectively. A preferred embodiment of actual time map information is illustrated in FIG. 16.

Additional data, regarding multimedia data recorded on the first layer, is recorded on a second layer. The additional data defines a reproduction unit for appropriately reproducing the multimedia data on the first layer. Here, the reproduction unit is divided into record units (or units of storage), which are described in the binary table format, or preferably, using markup language, and stored as description.xml. Navigation information, which is to be added to the additional data, is recorded on a third layer as menu.xml. The stored navigation information determines selection and sequences of data reproduction by controlling the reproduction unit recorded on the second layer.

In general, a menu screen is organized in a recording medium, on which a plurality of titles or chapters are recorded, to allow random access to a particular one of a plurality of titles or chapters for a user selection and immediate reproduction at a specified position. In the menu screen, still images and buttons are generally formed. Also, background music may be reproduced. When a user selects a button, the function associated with the button is executed. Referring to FIG. 14, the still images

and music data that constitute the menu screen are recorded as file3.jpg and file4.mp3 on the first layer.

The additional data recorded on the second layer describes information regarding the data recorded on the first layer. Video object data is multiplexed in a data stream to synchronize video, audio, and graphic data. The attributes of record units for video object data are described as description.xml, using the attributes of video, audio, and graphic data, and the numbers of audio, video, and graphic data. Further, the additional data provides information regarding reproduction units that are generated by combination or selection of record units.

In a preferred embodiment of the present invention, 'clip' is used as a record unit and 'cells', 'chapters', and 'titles' are used as reproduction units. The following description provides explanations on these units (see FIG. 15):

(1) clip: A clip is an object described in relation to the recording of multimedia data. FIG. 15 illustrates two video object clips. A video object clip has information on the time and position. Data belonging to a clip can be continuously reproduced. That is, an mpg file containing video object data and a time map file containing temporal position information are combined to form a clip. The time map file contains additional information that enables quick search for a desired temporal position of video object data when the video object is VBR encoded. That is, if a video object file is VBR encoded as shown in FIG. 17, the position of video object data at a time gap position, designated by predetermined time intervals from a starting position, is recorded in he table format shown in FIG. 16. If the data position in the table is called an entry, the total number of entries and time gaps may be recorded at the beginning of the table. The search for a desired position of data, with respect to a predetermined instant of time, can be accomplished by detecting the time gap position most proximate to the predetermined instant of time recorded in the table. Desired data can be precisely reproduced by reading data starting from the detected position. If data is recorded at a constant bit rate (CBR), it is possible to maintain the coded data amount, generated for a predetermined time. In this case, a clip can be constructed with only video object

data since it is possible to detect the desired position of data using time calculations according to the CBR without time map information.

The following information is used to define a video object clip consisting of video data, a plurality of audio data groups, and a plurality of graphic data groups:

(a) general attributes: clip identification information, video object clip files, and clip temporal position information file (time map file);

(b) video: screen size (e.g., 1920×1080, 1280×720, and 720×480), average bit rate (e.g., 4M, 6M, 10M, and 20M), screen output rate (e.g., 60Hz, 30Hz, and 24Hz), scanning types (e.g., progressive and interlaced types);

(c) audio: audio stream identification information, audio encoding information, linguistic attributes of each audio data group (e.g., Korean, and English), and application attributes of each audio data group (e.g., main audio, sub audio, and commentary); and

(d) graphic: graphic stream identification information, graphic encoding information, linguistic attributes of each graphic data group (e.g., Korean, and English), application attributes of each graphic data group (e.g., subtitles, and animation)

The above information, however, may be changed according to the recoding type, encoding type, and multiplexing type of multimedia data. Also, a record unit may form a hierarchical structure, and thus, a record unit subordinate to a clip may be present. For convenience, in the present invention, a record unit is determined to be made of clips.

In general, a reproduction unit has a hierarchical structure, that is, the reproduction unit consists of a plurality of reproduction sub units. In this case, a reproduction sub unit is defined as a unit of a reproduction sequence or a point of random access.

(2) cell: A cell is a reproduction unit that is described in relation to the reproduction of multimedia data, each cell designating a clip or a portion of a clip. The

cell represents the lowest layer of a reproduction unit. That is, a cell, i.e., a reproduction unit, is linked to at least one clip, i.e., a record unit, in the reproduction of multimedia data.

The cell is defined using identification information, and starting and ending times of the related clip. However, if the cell designates a whole clip, not a portion thereof, starting and ending times are not additionally specified.

(3) chapter: A chapter is a reproduction unit described in relation to the reproduction of data, each chapter consisting of at least one cells. In general, a chapter is defined by a chapter name, chapter identification information, and information regarding at least one cell belonging to the chapter.

In the present invention, a chapter can be understood as a reproduction point which allows a user to perform random access. Thus, within a title, a user can search for or reproduce desired data in units of chapters. In general, a menu screen provides a menu in which desired data can be selected in units of chapters.

(4) title: A title consists of a plurality of chapters. A plurality of titles may be stored in a storage medium and can be sequentially reproduced according to the sequence of title identification signs. In general, a menu screen provides a menu which allows selection of a title, the title defined by a title name, a title identification sign, and information regarding at least one chapter belonging to the title.

FIG. 15 illustrates the relationship between video object data recorded on the first layer, and clips, chapters, and titles recorded on the second layer.

In the present invention, information regarding the second layer is described in two types of units, i.e., record units and reproduction units, while data regarding navigation is eliminated. In general, the data is described in a binary table format or through a markup language. In the present invention data is described using a markup

language, since the markup language is more advantageous than the binary table. Extensible markup language (XML), a representative example of a markup language, is defined in W3C, which prescribes recommendations for the Internet. With XML, it is possible to describe a variety of databases and documents. When information regarding the second layer is described with XML, it is very easy to ensure extendibility and backward compatibility.

Hereinafter, an XML-based language will be defined describing additional information to be stored in the second layer, according to a preferred embodiment of the present invention. The XML-based language is referred to as the media description language (MDL). XML is described with a combination of hierarchical elements. Also, each element may have several attributes. The name of an element is described within the sign '<>'. The regulations for describing a sub element are mentioned below.

MDL, according to a preferred embodiment of the present invention, has elements and attributes as described below. Here, a document is understood as a unit of data that is described and stored utilizing the markup language.

(a) The uppermost element of a MDL document is described using <mdl> or an equivalent value.

(b) The element <mdl> may have the following sub elements:

- <head>
- <body>

(c) The element <head> contains all information regarding a storage unit and may have the following sub element:

- <meta>

(d) The element <meta> defines a blank element in which the features of a document are defined and appropriate values are allocated to the features. Each <meta> element denotes a pair including an attribute and a value.

The attributes of elements defined in the element <meta> are as follows:

- name: A name denotes a document feature defined in the element <meta>. The name is an attribute indispensable to the element <meta>.

- content: A content denotes a feature value defined in the element <meta>. The content is also an attribute indispensable to the element <meta>.

The following are examples of the element <meta> excluding conventional examples of the element <meta>:

e.g., (i) <meta name="type" content="mdl-disc" />: this element describes a disc that is manufactured using the media description language; and

(ii) <meta name="region" content="1" />: this element indicates that the regional code of the disc is 1.

(e) The element <body> may have the following sub elements:

- <clip>
- <title>

(f) The element <clip> may have the following attributes:

- id: *id* represents the identification information for each clip. *id* varies according to the type of storage medium (ex. 1, 2, or 3)
- src: *src* represents a video object data file such as 'file1.mpg'
- tmap_src: *tmap_src* represents a file containing a time map table (ex. file1timemap.dat)

The element <clip> may have the following sub elements:

- <video>
- <audio>
- <graphic>

(g) The element <video> may have the following attributes:.

- resolution : image sizes in the vertical and horizontal directions, e.g., 1920×1080, or 1280×720
- frame_rate : the number of frames output per second, e.g., 60, 30, 24, or 50

- scanning : scanning indicates whether an image is sequentially scanned or not, i.e., whether the image is progressively scanned or interlaced scanned.
- bit_rate : average bit rate, e.g., vbr, 4m, 6m, 8m, 10m, or 20m
- stream_id : stream id of an MPEG PES stream, e.g., 0xe0

(h) The element <audio> may have the following attributes:

- encoding: *encoding* represents an encoding method, e.g., mp1, mp2, mp3, ac3, lpcm, or dts
- sampling_rate: sampling rate, e.g., 48k, 96k, or 192k
- quantization_bit: the number of quantized bits, e.g., 16, or 24
- bit_rate: an encoded bit rate, e.g., vbr, 128k, or 384k
- channel_no: total number of channels, e.g., 2, 5, or 7
- language : linguistic attributes, e.g., none, en, ko, jp, or fr
- application: usage of an audio stream, e.g., main, sub, or commentary
- stream_id : stream ID of an MPEG PES stream, e.g., 0xc0, or 0툥
- sub_stream_id : sub stream ID of an MPEG PES stream, e.g., none, 0x80, or 0xa0

(i) The element <graphic> may have the following attributes:

- encoding: *encoding* represents an encoding method, e.g., dvd_subpicture
- application: usage of s graphic stream, e.g., animation or sub_title
- language: linguistic attributes, e.g., none, en, ko, jp, or fr
- stream_id : Stream ID on an MPEG PES stream, e.g., 0xBD
- sub_stream_id : Sub stream ID on an MPEG PES stream, e.g., 0x20

(k) The element <title> may have the following attributes:

- name : title name, e.g., White Snow
- id: title identification information that varies according to the type of storage medium, e.g., 1, 2, or 3

Also, the element <title> may have the following sub elements:

- <chapter>

(l) The element <chapter> may have the following attributes:

- name : chapter name, e.g., Dwarf
- id : chapter identification information that varies according to a title, e.g., 1, 2, or 3

Also, the element <chapter> may have the following sub element:

- <cell>

(m) The element <cell> may have the following attributes:

- clip_id : identification number of a clip to which the cell is linked
- start_time : starting time in a clip *clip_id*
- end_time : ending time in a clip *clip_id*

A summary of the above attributes and elements is shown in Table 1.

[Table 1]

Element	Attribute	Sub Element
<mdl>		<head>/<body>
<head>		<meta>*
<meta>	name, content	EMPTY
<body>		<clip>+/<title>+
<clip>	id, src, tmap_src	<video>/<audio>*/<graphic>*
<video>	resolution, frame_rate, scanning, bit_rate, stream_id	EMPTY
<audio>	encoding, sampling_rate, quantization_bit, bit_rate, channel_no, language, application, stream_id, sub_stream_id	EMPTY
<graphic>	encoding, application, language, stream_id, sub_stream_id	EMPTY

<title>	name, id	<chapter>+
<chapter>	name, id	<cell>+
<cell>	clip_id, start_time, end_time	EMPTY

The additional data of FIG. 15, according to a preferred embodiment of the present invention, may be described using the above elements and attributes, as follows:

```
<?xml version="1.0"?>
<!DOCTYPE m0|
        PUBLIC "http://mdl-disc.org/mdl1/mdl.dtd">

<mdl>
<head>
        <meta name="type" content="mdl-disc" />
        < meta name="title" content="White Snow Special" />
        <meta name="region" content="1" />
</head>

<body>
        <clip id="1" src="file1.mpg" tmap_src="file1tmap.dat">
                <video resolution="1920x1080" frame_rate="24"
                scanning="progressive" bit_rate="vbr" stream_id="0xe0" />
                <audio encoding="ac3" sampling_rate="48k"
quantization_bit="16"                                   bit_rate="384k"
channel_no="5" language="en"
application="main" stream_id="0xbd" sub_stream_id="0x80" />
                        <audio encoding="ac3" sampling_rate="48k" quantization_bit="24"
                                        bit_rate="384k" channel_no="5" language="ko"
```

```
                                                    application="main"
stream_id="0xbd" sub_stream_id="0x81" />
                    <audio encoding="ac3" sampling_rate="48k" quantization_bit="24"
                                                    bit_rate="384k" channel_no="5" language="jp"
                                                application="main" stream_id="0xbd"
sub_stream_id="0x84" />
                    <audio encoding="mp1" sampling_rate="48k" quantization_bit="16"
                                                    bit_rate="384k" channel_no="2" language="en"

application="commentary" stream_id="0xc0"
                sub_stream_id="none" />
            <graphic encoding="dvd_subpicture" application="sub_title"
                                language="en" stream_id="0xbd" sub_stream_id="0x20" />
                <graphic encoding="dvd_subpicture" application="sub_title"
                                                language="ko" stream_id="0xbd"
sub_stream_id="0x21" />
                    <graphic encoding="dvd_subpicture" application="sub_title"
                                                    language="jp" stream_id="0xbd"
sub_stream_id="0x22" />
                </clip>
                ...
                <clip id="5" src="file5.mpg" tmap_src="filek5map.dat">
                    <video resolution="1920x1080" frame_rate="24"
                                            scanning="progressive" bit_rate="vbr"
stream_id="0xe0" />
                    <audio encoding="ac3" sampling_rate="48k" quantization_bit="16"
                                                bit_rate="384k" channel_no="5" language="en"

application="main" stream_id="0xbd" sub_stream_id="0x80" />
```

```
                    <audio encoding="mp1" sampling_rate="48k" quantization_bit="16"
                                                bit_rate="384k" channel_no="2" language="en"

application="commentary" stream_id="0xbd"
                sub_stream_id="0x83" />
                </clip>

                <title id="1" name="White Snow">
                <chapter id="1" name="Mirror">
                        <cell clip_id="1" start_time="0:00:00"
end_time="0:05:00"/>
                        </chapter>
                        <chapter id="2" name="7 Dwarves">
                                <cell clip_id="1" start_time="0:05:00" end_time="0:20:00"/>
                                <cell clip_id="2" start_time="0:00:00" end_time="1:00:00"/>
</chapter>
                         ...
                </title>
                <title id="2" name="White Snow Supplementary">
                        <chapter id="1" name="How to make Snow White">
                        <cell clip_id="5" start_time="0:00:00" end_time="0:20:00"/>
                        </chapter>
        </title>
        </body>
        </mdl>
```

In the embodiment, actual video object data consists of five clips represented in record units, and two titles represented in reproduction units.

In the case of VBR video object data, a clip is described as time position information data, which constitutes a portion of time map information. Thus, in the embodiment, each clip is described using the two attributes *src* and *tmap_src*, such that the clip data and titles are linked to each other. In addition, each clip contains many attributes of video, audio, and graphic data, and is referred to prior to data reproduction.

Title 1 is a subject title consisting of a plurality of chapters. The first chapter is comprised of a cell linked to a portion of clip #1. Thus, only the portion of clip #1 is reproduced during reproduction of the first chapter. The second chapter is comprised of two cells. The first cell designates reproduction of data after time 0:05:00 of clip #1. Accordingly, in order to start reproduction from the second chapter, the position of desired data must be detected using time map information, specifically, the time position information regarding clip #1.

As shown in its title name attribute, title 2 is comprised of an additional image, i.e., a supplementary image. Title 2 is recorded as clip #5.

As mentioned above, improved extensibility can be achieved by describing information regarding the second layer using the markup language. In other words, an information structure of a new concept can be described using new elements or attributes. Then, the existing reproducing apparatuses can reproduce data using the existing information rather than newly generated information. That is, it is possible to maintain the reproduction of data regarding the existing element using the existing reproducing apparatus. Let us assume that a new element <bookmark> is added to the <title> element and the following information is recorded on the second layer:

```
<title id="1"... >
<bookmark point="1:00:00" />
<chapter id="1" ..>
..
</title>
```

The element <bookmark> is an information structure that is newly defined to allow a direct access to a particular position of a title. When the storage medium, on which the element <bookmark> is recorded, is inserted into a conventional reproducing apparatus, it is possible to reproduce data recorded on the storage medium using title and chapter information. However, bookmark information is ignored because it cannot be reproduced using the existing reproducing apparatus.

As mentioned above, multimedia data can be reproduced using only data recorded on the first and second layers. In this case, a reproducing apparatus reads the information recorded on the second layer, and determines the format of recorded data recorded, title, and the number of chapters belonging to each title. Next, the reproducing apparatus informs a user of the detection result through an appropriate user interface, receives user input, and reproduces a desired reproduction unit. The user interface includes a user output device 300 and a user input device 400 as shown in FIG. 1. In detail, the user output device 300 is an apparatus, such as a television (TV), that outputs multimedia data, and the user input device 400 is an apparatus, such as a remote control, that receives user input. The user interface, according to a preferred embodiment of the present invention, consists of a chapter menu in a menu screen, which allows selection of chapters belonging to each title. The user selects a title number or chapter number from each menu using a remote control, resulting in detection and reproduction of a desired reproduction unit.

A storage medium, according to the present invention, further includes navigation information recorded on the third layer. Navigation, which is similar to a conventional menu, includes selection using a user input device such as a remote control and reproduction of a reproduction unit to a user output device, such as a TV. Also, navigation may include control of the following data reproduction using a current state of data reproduction. Here, the recording apparatus is as shown in FIG. 1. Also, sequences of data reproduction may be determined differently. For instance, it is preferable that parental level control is performed during data reproduction, that is,

sequences of data reproduction are determined depending on whether a user is an adult or a juvenile.

For this end, a reproducing apparatus may be manufactured as show in FIG. 18. Referring to FIG. 18, a playback engine denotes a block for processing data recorded on the first and second layers. If a storage medium does not contain data in its third layer, a presentation & navigation engine of FIG. 18 converts user input into an Application Program Interface (API), which can be recognized by the playback engine, and provides the API to the playback engine. If the user input is key input recognized by the playback engine, the user input is directly transmitted to the playback engine.

The data recorded on the third layer includes presentation data, which is used to arrange a menu screen or a reproduction unit in a screen, and navigation data, which is used to select a reproduction unit in response to user input or control data reproduction, according to the state of a certain playback engine. Presentation data, according to a preferred embodiment of the present invention, is described using HyperText Markup Language (html) or Extensible HyperText Markup Language (xhtml). Also, navigation data may be described with a Script language or a markup language capable of describing timing and synchronizing. A typical example of a Script language is Java script that is interpreted and executed in units of lines, and a typical example of the markup language having timing and sync definitions is synchronized markup interface language (SMIL).

A navigation engine performs navigation by controlling a reproduction unit recorded on the second layer, according to a user's selection or an event generated by the playback engine.

The following describes methods of laying out reproduction units, recorded on the second layer, in a screen using XHTML and JavaScript, and controlling navigation, according to a preferred embodiment of the present invention.

An example of controlling navigation with a markup document, which allows a user input key to be directly transmitted to the playback engine, is as follows:

- Markup Document 1

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html
     PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
  <head>
     <title>White Snow</title>
    <meta name="Default_audio" content="0x80" />
    <meta name="Default_subtitle" content="0x20" />
    <script type="text/javascript">
   function RcKeyEventHandler(evt)
   {
       if(evt.keyCode == 10)
           MDLvideo.InputRCKey(10);
       if(evt.keyCode == 11)
           MDLvideo.InputRCKey(11);
       if(evt.keyCode == 12)
           MDLvideo.InputRCKey(12);
       if(evt.keyCode == 13)
           MDLvideo.InputRCKey(13);
       if(evt.keyCode == 14)
           MDLvideo.InputRCKey(14);
       if(evt.keyCode == 15)
           MDLvideo.InputRCKey(15);
       if(evt.keyCode == 16)
           MDLvideo.InputRCKey(16);
       if(evt.keyCode == 17)
           MDLvideo.InputRCKey(17);
```

```
        if(evt.keyCode == 18)
            MDLvideo.InputRCKey(18);
     }
    </script>
    <script type="text/javascript">
    function setupEventHandler()
    {
    bodyNode.addEventListener("rckeypress",RcKeyEventHandler,true);
    }
    </script>
        </head>

        <body id="bodyNode" onload="setupEventHandler()">
       <object data="MDLvideo;" />
    </body>
    </html>
```

A key event interface of a user input device is as follows:

```
interface RCKeyEvent : UIEvent {
const unsigned long VK_0                    = 0; // keyCode is "0", keyName is "0"
const unsigned long VK_1                    = 1; // "1"
const unsigned long VK_2                    = 2; // "2"
const unsigned long VK_3                    = 3; // "3"
const unsigned long VK_4                    = 4; // "4"
const unsigned long VK_5                    = 5; // "5"
const unsigned long VK_6                    = 6; // "6"
const unsigned long VK_7                    = 7; // "7"
const unsigned long VK_8                    = 8; // "8"
```

```
const unsigned long VK_9                = 9; // "9"
const unsigned long VK_PLAY_PAUSE       = 10; // "PlayPause"
const unsigned long VK_STOP                     = 11; // "Stop"
const unsigned long VK_FF               = 12; // "FastForward"
const unsigned long VK_FR               = 13; // "FastRewind"
const unsigned long VK_SKIP_PREV        = 14; // "Prev"
const unsigned long VK_SKIP_NEXT                = 15; // "Next"
const unsigned long VK_SUBTITLE         = 16; // "Subtitle"
const unsigned long VK_AUDIO                    = 17; // "Audio"
const unsigned long VK_MENU                     = 18; // "Menu"
const unsigned long VK_UP               = 19; // "Up"
const unsigned long VK_LEFT                     = 20; // "Left"
const unsigned long VK_RIGHT                    = 21; // "Right"
const unsigned long VK_DOWN                     = 22; // "Down"
const unsigned long VK_OK               = 23; // "OK"
const unsigned long VK_RETURN           = 24; // "Return"
const unsigned long VK_EXIT             = 25; // "Exit"
readonly attribute unsigned long keyCode;
readonly attribute DOMString keyName;
void initRCKeyEvent (in DOMString typeArg,
in boolean canBubbleArg,
in boolean cancelableArg,
in unsigned long keyCode,
in unsigned long keyName);
}
```

The above example of the markup document reveals that "Markup Document 1" is largely comprised of layout information, which is related to a video object displayed by a markup document, and a script, is used to control reproduction data recorded on the

second layer in response to user input. Also, the key event interface of the user input device defines an interface that allows key values, used in the user input device, to be used in a document object model (DOM).

First, the above example of a markup document includes sequentially a declaration that enables use of XHTML and an element <head>, in which an element <title>, an element <meta>, and an element <script> are included. In general, the element <title> represents the title subject of the markup document, and the element <meta> indicates default audio and subtitles contained in the video object, which is reproduced in the markup document.

Also, event registration information according to user input is described using JavaScript language, as follows:

```
<script type="text/javascript">
function setupEventHandler()
{
bodyNode.addEventListener("rckeypress",RcKeyEventHandler,true);
}
</script>
```

The event registration information indicates that a function *RcKeyEventHandler* is called when an event *rckeypress* occurs, i.e., a user presses a key of a user input device.

Also, event processing information is described using the JavaScript language as follows:

```
<script type="text/javascript">
 function RcKeyEventHandler(evt)
 {
     if(evt.keyCode == 10)
```

```
            MDLvideo.InputRCKey(10);
        if(evt.keyCode == 11)
            MDLvideo.InputRCKey(11);
        if(evt.keyCode == 12)
            MDLvideo.InputRCKey(12);
        if(evt.keyCode == 13)
            MDLvideo.InputRCKey(13);
        if(evt.keyCode == 14)
            MDLvideo.InputRCKey(14);
        if(evt.keyCode == 15)
            MDLvideo.InputRCKey(15);
        if(evt.keyCode == 16)
            MDLvideo.InputRCKey(16);
        if(evt.keyCode == 17)
            MDLvideo.InputRCKey(17);
        if(evt.keyCode == 18)
            MDLvideo.InputRCKey(18);
    }
</script>
```

The event processing information indicates that *MDLvideo.InputRCKey(10)* is executed when a key code *RcKeyEventHandler* is 10. Here, the object *MDLvideo* performs data reproduction using additional data stored in the second layer, that is, the object *MDLvideo* corresponds to the playback engine. In other words, if a code allocated to a key is 10 and a user presses this key, the presentation & navigation engine sends a control command to the playback engine using a command *API MDLvideo.InputRCKey*. In response to this command, the playback engine performs the reproduction control operation allocated to the key code 10, i.e., reproduces or temporarily stops reproduction of a video object.

The object *MDLvideo* is embedded in the element <body> of the markup document using the element <object>. Meanwhile, it is possible to embed a plurality of objects in the element <body> of the markup document. In this case, it is preferable that the layout of the markup document uses cascading style sheet (CSS).

Next, a method of providing a specific functional command to the playback engine and performing navigation will be described. FIG. 19 illustrates a method of forming a menu screen for navigation.

Image and text data are described as presentation data for display on a screen. It is preferable that the screen is composed of text data described with xhtml, or image data recorded on the first layer. Referring to FIG. 19, four buttons, i.e., title 1, title 2, title 3, and return, are displayed on the screen. Image or text data forming the four buttons is described using xhtml. In the screen, a user can select and press one of these buttons. More specifically, the user applies directional keys of a user input device to select one of these buttons and applies the OK key to press the selected button. Alternatively, the screen may be constructed such that a certain operation is executed when the user presses a certain button using the keys of the user input device as access keys. When the user selects one of these buttons, the selected button is highlighted. The above functions are provided by a presentation module of the presentation & navigation engine. It is preferable that the presentation module is an xhtml browser. If the user presses one button, the operation connected to the button is performed.

Likewise, a command is provided to the playback engine in order to reproduce multimedia data using data recorded on the first and second layers. That is, the presentation & navigation engine provides the playback engine with the command for controlling a reproduction unit recorded on the second layer.

- Markup Document 2

```
<?xml version="1.0" encoding="UTF-8"?>
```

```
<!DOCTYPE html
      PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
   <title>White Snow</title>
   <meta name="Default_audio" content="0x80" />
   <meta name="Default_subtitle" content="0x20" />
    </head>
    <body>
    <table>
       <tr>
         <td><img src="file1.jpg" alt="title1"/></td>
         <td><button type="button" name="title 1"
onclick="MDLvideo.playTitle(1)"/></td>
       </tr>
       <tr>
         <td><img src="file2.jpg" alt="title2"/></td>
         <td><button type="button" name="title 2"
onclick="MDLvideo.playTitle(2)"/></td>
       </tr>
       <tr>
         <td><img src="file3.jpg" alt="title3"/></td>
         <td><button type="button" name="title 3"
onclick="MDLvideo.playTitle(3)"/></td>
       </tr>
       <tr>
         <td><button type="button" name="Return" onclick="MDLvideo.play()"/></td>
       </tr>
</body>
```

</html>

Markup Document 2 represents the menu screen illustrated in FIG. 19. Referring to Markup Document 2, commands for controlling reproduction units recorded on the second layer are described in an 'onclick' event of each button. When a button for reproducing title 1 is clicked, a command *MDLvideo.playTitle(1)* is sent to the navigation engine. Then, the navigation engine provides the commands and parameters to the playback engine for reproduction of corresponding data. For instance, *playTitle* is a title reproduction parameter that indicates title number. A function of providing such a command is called a method.

Examples of the method are as follows:

method for an object *MDLvideo* {
playTitle(title_no): reproduce data starting from a title indicated in title_no.
playTime(time): reproduce data starting from indicated time.
pause(): pause data reproduction
stop(): stop data reproduction
play(): start data reproduction
forward_scan(speed): high-speed forward data reproduction at a speed indicated in (speed)
backward_scan(speed): high-speed backward data reproduction at a speed indicated in (speed)
playChapter(chapter_no): when chapter_no == 0, move to chapter having the largest chapter ID and reproduce data of the chapter. If chapter_no > Maximum chapter ID, move to a chapter having the smallest chapter ID and reproduce data of the chapter.
}

If specific conditions required by the playback engine are satisfied, an event for processing data recorded on the third layer is generated in the presentation & navigation engine. For instance, whenever a chapter begins, an event is generated and provided to the presentation & navigation engine, and then, the presentation engine displays the start of the chapter in a screen. Information regarding an event which is provided to the presentation & navigation engine and registered to the playback engine, may also be recorded on the third layer.

- Markup Document 3

```
<smil>
  <head>
    <meta name="title" content="White Snow"/>
    <meta name="Default_audio" content="0x80" />
    <meta name="Default_subtitle" content="0x20" />
    <layout>
      <root-layout width="1920" height="1080"/>
      <region id="img_scene1" top="45" left="30" width="600" height="300"/>
      <region id="img_scene2" top="390" left="30" width="600" height="300"/>
      <region id="img_scene3" top="735" left="30" width="600" height="300"/>
      <region id="img_title_button1" top="95" left="660" width="600" height="200"/>
      <region id="img_title_button2" top="440" left="660" width="600" height="200"/>
      <region id="img_title_button3" top="785" left="660" width="600" height="200"/>
      <region id="img_return" top="885" left="1290" width="600" height="150"/>
    </layout>
  </head>
  <body>
    <par>
      <img src="file1.jpg" region="img_scene1"/>
      <img src="file2.jpg" region="img_scene2"/>
```

```
        <img src="file3.jpg" region="img_scene3"/>
        <img src="file4.jpg?url=command:MDLvideo.playTitle(1)"
region="img_title_button1"/>
        <img src="file5.jpg?url=command:MDLvideo.playTitle(2)"
    region="img_title_button2"/>
        <img src="file6.jpg?url=command:MDLvideo.playTitle(3)"
    region="img_title_button3"/>
        <img src="file7.jpg?url=command:MDLvideo.play()" region="img_return"/>
       </par>
    </body>
    </smil>
```

Markup Document 3 represents data that is stored in the third layer and described using SMIL, which is a markup language with timing and synchronization functions. SMIL is also largely divided into the elements <head> and <body>. In Markup Document 3, the element "head" includes elements "meta" and "layout". The element "meta" has already been described with respect to the above markup documents and its description will be omitted here. The element "layout" has children elements "root-layout" and "region". The element "root-layout" describes the size and background color of a document to be displayed. The element "region" describes the layout of a SMIL document region where each media clip is formed, and ID of the respective region.

The element "body" includes the following elements, which describes media clip sources:

<animation ...../> - Shockwave Flash File (.swf)

<audio ...../> - Audio Clip File (.rm)

 - Image Clip File (.jpg .gif) / Not Use Animation GIF

<ref ...../> - Document of Different Use (.htm .rp)

<text ...../> - Text (.txt)

<textstream ..../> - Text Stream (.rt)

<video ...../> - Video Clip (.rm)

In Markup Document 3, the element "img" is used to create the menu screen shown in FIG. 19. The element "img" can be used to link a document to a desired image or insert a command into the document. Also, Markup Document 3 may further include elements "a" and "anchor" for use as a hyperlink. In Markup Documents, all images are linked except three images. In SMIL, a reproduction control attribute "url" in the element "img" may have the following attribute values, including *playTitle*:

url : hyperlink URL

command:MDLvideo.playTitle(title_no): reproduce data starting from a title indicated in title_no

command:MDLvideo.playTime(time): reproduce data starting from time indicated in (time)

command:MDLvideo.play(): start data reproduction

command:MDLvideo.stop(): stop data reproduction

command:MDLvideo.pause(): pause data reproduction

command:MDLvideo.forward_scan(speed): high-speed forward reproduction at a speed specified in (speed)

command:MDLvideo.backward_scan(speed): high-speed backward reproduction at a speed specified in (speed)

command:MDLvideo.playChapter(chapter_no): if chapter_no == 0, move to a chapter having the largest chapter ID and reproduce data of the chapter. If chapter_no > Maximum chapter ID, move to a chapter having the smallest chapter ID and reproduce data of the chapter.

As described above, according to a storage medium of the present invention, information regarding reproduction of multimedia data is divided into record units and reproduction units, and additional information regarding the record units and reproduction units are recorded on a second layer. Also, information regarding selection and navigation of a desired reproduction unit is recorded on a third layer. In this way, the operation of each layer can be distinguished. In the second layer, data is recorded using a markup language to improve extensibility. In the third layer, data is also recorded using a markup language for creating additional data, representing a menu screen or the layout of a reproduction unit. Data regarding selection of a reproduction unit and the reproduction sequence is described using a script language or markup language with timing and synchronization functions.

In order to store multimedia data in a storage medium, a multimedia data recording apparatus, according to the present invention, records the multimedia data on the first layer of the storage medium; divides additional information, regarding multimedia recorded on the first layer, into record units and reproduction units; and records the divided additional information on the second layer in a table format or by using a markup language. Here, the record units and the reproduction units may have multiple hierarchical structures. According to a preferred embodiment of the present invention, the record unit is a clip that is made by linking time information to position information when video object data is recorded at VBR, and the reproduction units are cells linked to a clip or a portion of a clip, chapters linked to a plurality of cells, and titles linked to a plurality of chapters.

In addition, navigation data regarding selection of a reproduction unit or the reproduction sequence is recorded on the third layer. The navigation data is described using either a script language executed in units of lines, or a markup language with timing and synchronization functions. Also, in the third layer, presentation data, representing a menu screen for selection of a reproduction unit or reproduction sequence, is described using a markup language.

While this invention has been particularly shown and described with reference to preferred embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the invention as defined by the appended claims.

[Effect of the Invention]

A storage medium, according to the present invention, contains multimedia data and additional data. The additional data contains two different types of data: additional information, which contains a record unit, attributes, and a reproduction unit of the multimedia data, and navigation information, which relates to selection of a reproduction unit and the reproduction sequence. The additional information is described using a markup language, thereby enabling addition of new multimedia data regardless of the extension of the standard. Also, even if a new record unit or reproduction is prescribed, it is easy to support implementation of the extension. Further, both or one of a markup language and a script language is used to describe navigation data, which represents selection of a reproduction unit or reproduction sequence. A markup language is also used to describe presentation data which represents a menu screen for selection of a reproduction unit and a screen layout for data reproduction, thereby enabling a menu structure and navigation with a high degree of flexibility.

What is claimed is:

1. A multimedia data storage medium in which multimedia data is stored, comprising:

a first layer in which multimedia data is stored; and

a second layer in which, when the multimedia data is divided into a record unit and a reproduction unit, information regarding attributes of the record unit and relationship between the record unit and the reproduction unit are described with a markup language using elements and attributes.

2. The medium of claim 1, wherein the multimedia data recorded on the first layer is one of video object data, still image data, or audio data.

3. The medium of claim 1, wherein if the multimedia data is video object data, the multimedia data is coded at a variable bit rate (VBR), and reproduction time and linkage information regarding a position of reproduction data, are described as temporal and position information in a table format and recorded on the first layer.

4. The medium of claim 3, wherein the record unit comprises a clip that is made by linking the video object data to the temporal and position information.

5. The medium of claim 3 or 4, wherein the reproduction unit comprises a cell that designates a record unit or a portion of the record unit.

6. The medium of claim 1 or 2, wherein the reproduction unit has two layers that are the cell defined in claim 5 and a chapter linked to a plurality of cells.

7. The medium of claim 6, wherein the reproduction unit has two layers that are the cell defined in claim 5 and a chapter linked to a plurality of cells.

8. The medium of claim 7, wherein the reproduction unit has three layers that are the cell, the chapter, and a title linked to chapters.

9. The medium of claim 1 or 8, wherein information described using a markup language contains information regarding elements of the clip, cell, chapter, and title.

10. The medium of any one of claims 1 through 8, wherein navigation data, which is used to control selection of the reproduction unit and reproduction sequence, is recorded on a third layer, in addition to the information recorded on the first and second layers.

11. A multimedia data storage medium in which multimedia data is stored, comprising:

a first layer in which multimedia data is stored;

a second layer in which, when the multimedia data is divided into a record unit and a reproduction unit, information regarding attributes of the record unit and relationship between the record unit and the reproduction unit are described in a table format; and

a third layer in which navigation data regarding selection of a reproduction unit and reproduction sequence is stored.

12. The medium of claim 10 or 11, wherein the navigation data stored in the third layer is described using a script language that is interpreted and executed in units of lines, the navigation data controlling selection of a reproduction unit and reproduction sequence.

13. The medium of claim 10 or 11, wherein the navigation data stored in the third layer is described using a markup language with timing and synchronization

functions, the navigation data controlling selection of a reproduction unit and reproduction sequence.

14. The medium of claim 13, wherein presentation data is described with a markup language and stored in the third layer, the presentation data containing information regarding layout of a menu screen and screen structure of a reproduction unit.

15. The medium of claim 13, wherein the navigation data is described as an interface for selecting and controlling the reproduction unit recorded on the second layer.

16. The medium of claim 13, wherein the navigation data contains information regarding registration and execution of an event generating specific conditions, when the reproduction unit is reproduced.

17. A multimedia reproduction apparatus that reproduces data stored in a multimedia data storage medium that includes a first layer containing multimedia data and a second layer, in which information regarding the attributes of a record unit and the relationship between the record unit and a reproduction unit are described with respect to elements and attributes using a markup language, when the multimedia data is divided into the record unit and the reproduction unit, the apparatus that reads the information regarding the record unit and the reproduction unit from the second layer, detects a desired position of data, which corresponds to a record unit from multimedia data recorded on the first layer and is selected by a user through a user interface, and reproduces data at the detected position.

18. The apparatus of claim 17, wherein a menu screen is provided to select the reproduction unit through the user interface and reproduce the selected

reproduction unit, when the reproduction unit is stored by at least two layers.

19. The apparatus of claim 17, wherein if the storage medium further contains navigation information regarding selection of the reproduction unit and reproduction sequence on a third layer, the reproduction unit selected by a user through the user interface is reproduced based on the navigation information.

20. A multimedia reproduction apparatus that reproduces data stored in a multimedia data storage medium that includes a first layer containing multimedia data, a second layer in which information regarding the attributes of a record unit and the relationship between the record unit and a reproduction unit are stored in a table format when the multimedia data is divided into the record unit and the reproduction unit, and a third layer containing navigation information regarding selection of the reproduction unit and the reproduction sequence, the apparatus reproducing the reproduction unit, which is selected by a user through a user interface and based on the navigation information.

21. The apparatus of claim 20, wherein if the navigation data recorded on the third layer is described using a script language which is interpreted and executed in units of lines, navigation is performed by executing the script language, the navigation data controlling selection of the reproduction unit and reproduction sequence.

22. The multimedia reproduction apparatus of claim 20, wherein if the navigation data recorded on the third layer is described using markup language with timing and synchronization functions, navigation is performed to carry out the timing and synchronization functions, the navigation data controlling selection of the reproduction unit and reproduction sequence.

23. The apparatus of claim 20, wherein if the storage medium contains

presentation data described using a markup language, the presentation data is read from the storage medium and output in the form of a menu screen representing a user interface, the presentation data composing a menu screen.

24. A multimedia data recording apparatus that records multimedia data on a storage medium, the apparatus recording multimedia data on a first layer of the storage medium, dividing the multimedia data into a record unit and a reproduction unit, recording information and linkage information regarding the record unit and the reproduction unit using markup language, and storing the information and linkage information on a second layer of the storage medium, the multimedia data including video object data, still image data, audio data, or graphic data.

25. The apparatus of claim 23, wherein navigation data is recorded on a third layer of the storage medium, the navigation controlling selection of the reproduction unit and the reproduction sequence.

26. A multimedia data recording apparatus that records multimedia data in a storage medium, the apparatus recording multimedia data on a first layer of the storage medium, dividing the multimedia data recorded on the first layer into a record unit and a reproduction unit, recording information and linkage information regarding the record unit and the reproduction unit on a second layer in a table format, and storing navigation data in a third layer, the multimedia data including video object data, still image data, audio data, or graphic data, and the navigation data controlling selection of the reproduction unit and reproduction sequence.

27. The apparatus of claim 26, wherein the navigation data is described using a script language that is interpreted and executed in units of lines.

28. The apparatus of claim 26, wherein the navigation data is described using

a markup language with timing and synchronization functions.

29. The apparatus of claim 26, wherein presentation data is recorded on a third layer using a markup language, the presentation data composing a menu screen for selection of the reproduction unit or screen layout for displaying the multimedia data.

FIG. 1

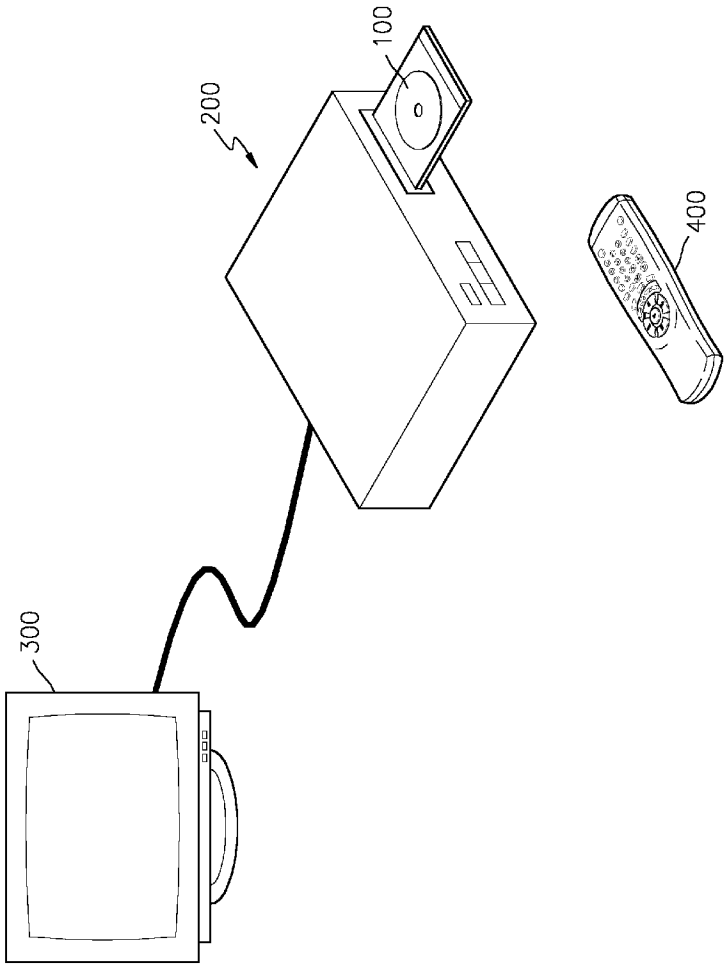


FIG. 2

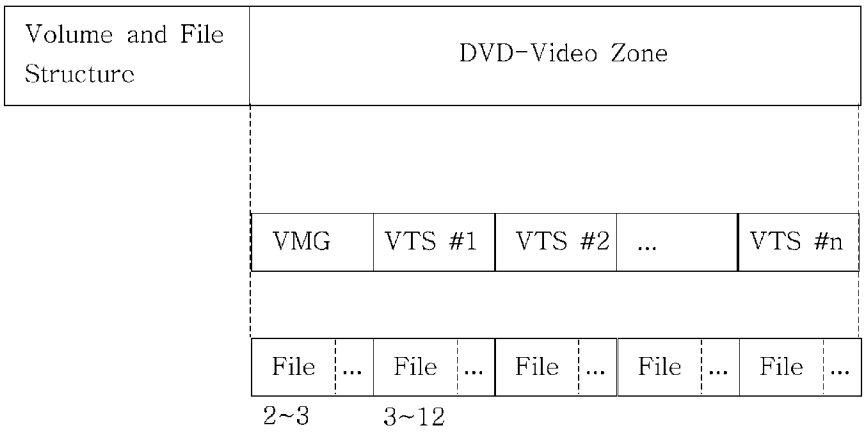
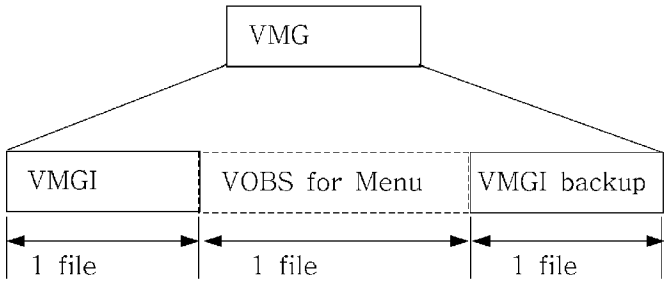

Volume and File Structure
DVD-Video Zone
VMG
VTS #1
VTS #2
...
VTS #n
File
...
File
...
File
...
File
...
File
...
2~3
3~12

FIG. 3


VMG
VMGI
VOBS for Menu
VMGI backup
1 file
1 file
1 file

5

FIG. 4

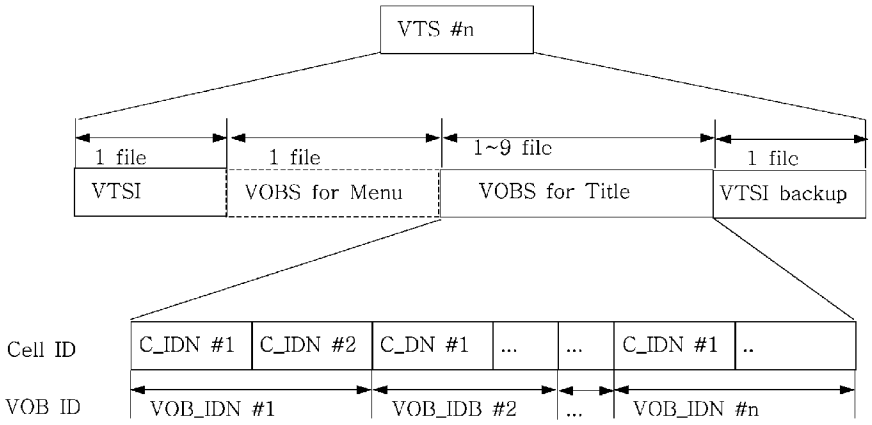


FIG. 5

FIG. 6

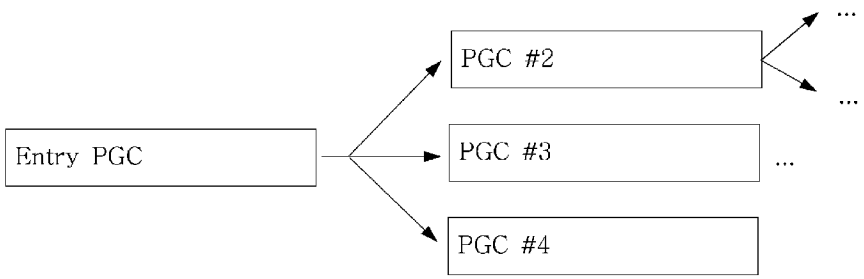


FIG. 7

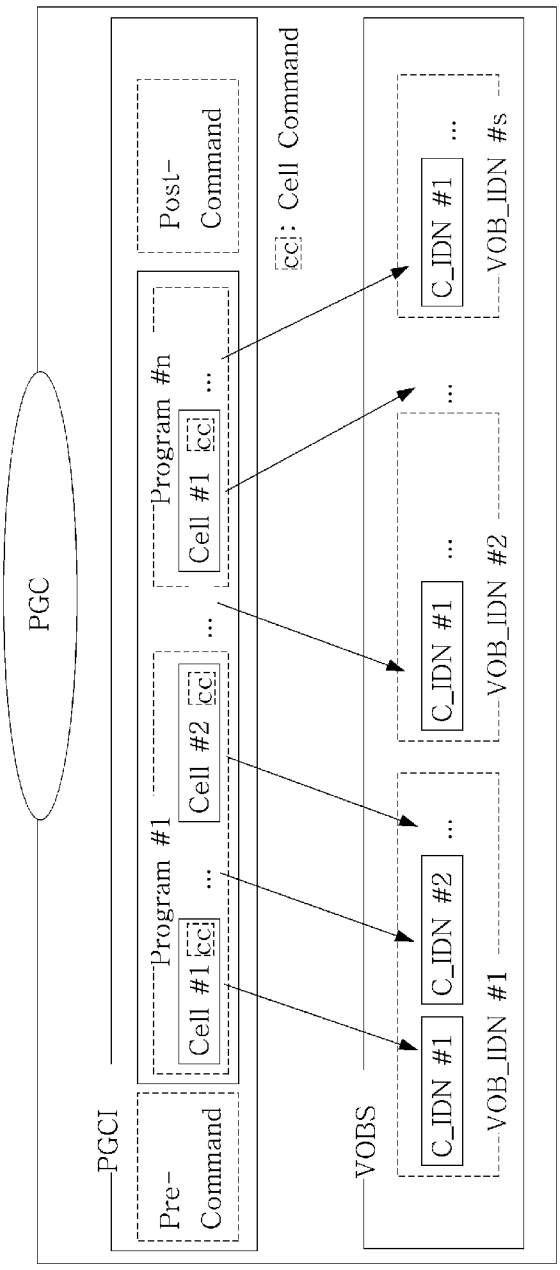

PGC
PGCI
Pre-Command
Program #1
Cell #1 cc
...
Cell #2 cc
...
Program #n
Cell #1 cc
...
Post-Command
cc: Cell Command
VOBS
C_IDN #1
C_IDN #2
...
VOB_IDN #1
C_IDN #1
...
VOB_IDN #2
C_IDN #1
...
VOB_IDN #s

FIG. 8

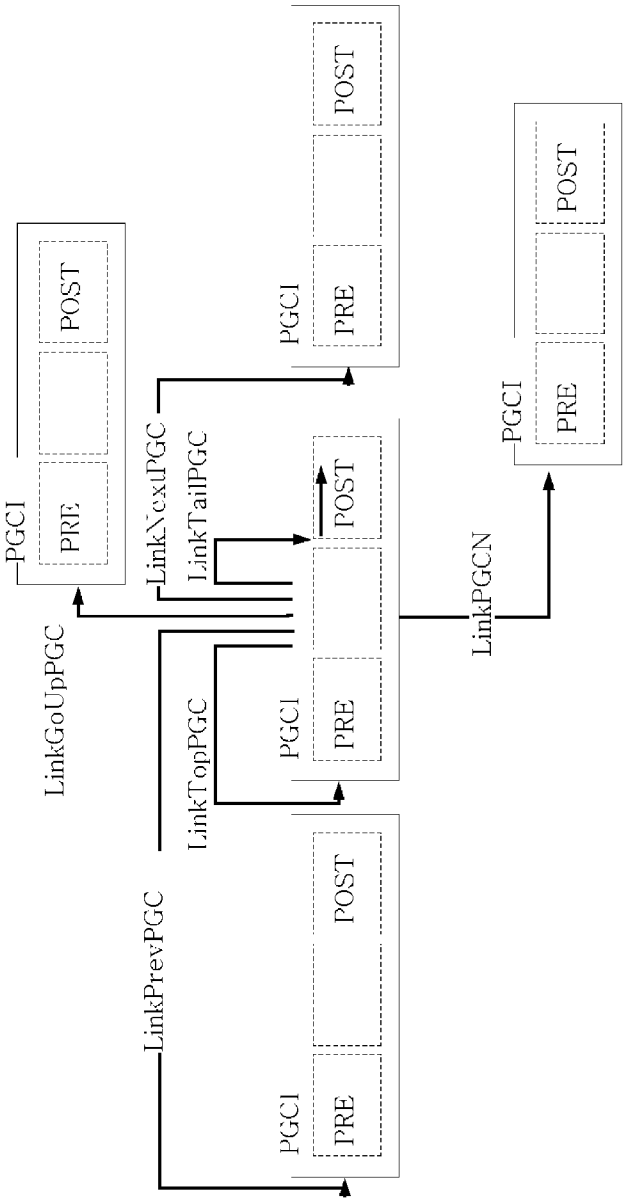

PGCI
PRE
POST
LinkGoUpPGC
LinkNextPGC
LinkTailPGC
LinkTopPGC
LinkPrevPGC
LinkPGCN

FIG. 9

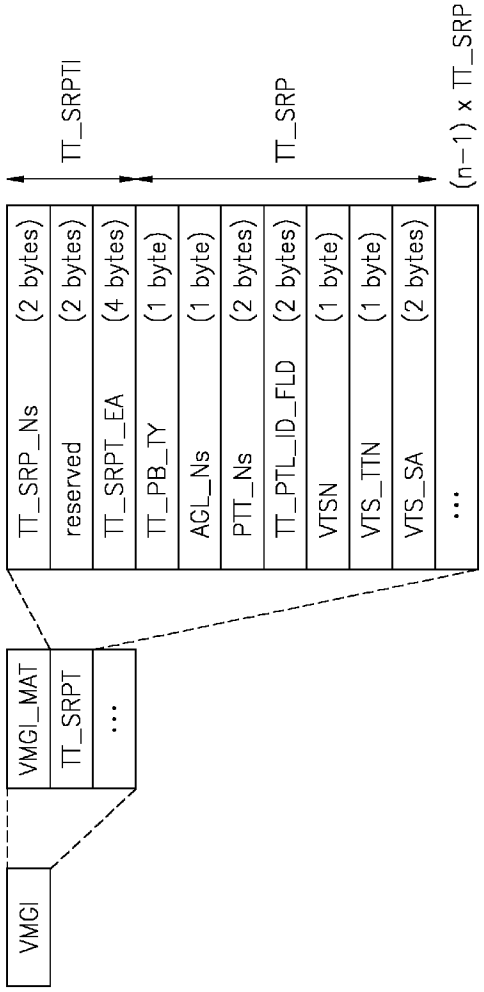

VMGI
VMGI_MAT
TT_SRPT
...
TT_SRP_Ns (2 bytes)
reserved (2 bytes)
TT_SRPT_EA (4 bytes)
TT_PB_TY (1 byte)
AGL_Ns (1 byte)
PTT_Ns (2 bytes)
TT_PTL_ID_FLD (2 bytes)
VTSN (1 byte)
VTS_TTN (1 byte)
VTS_SA (2 bytes)
...
TT_SRPTI
TT_SRP
(n-1) x TT_SRP

FIG. 10

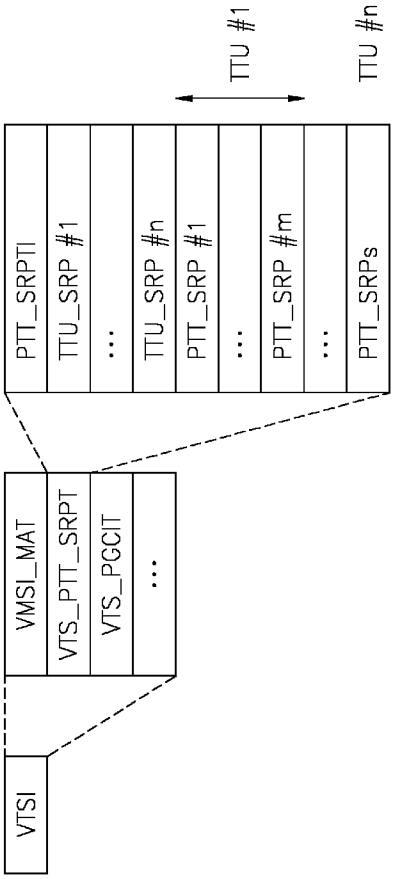

VTSI
VMSI_MAT
VTS_PTT_SRPT
VTS_PGCIT
...
PTT_SRPTI
TTU_SRP #1
...
TTU_SRP #n
PTT_SRP #1
...
PTT_SRP #m
...
PTT_SRPs
TTU #1
TTU #n

FIG. 11

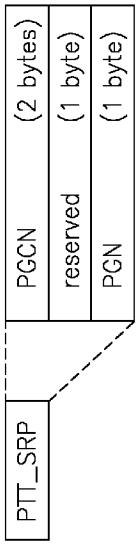

PTT_SRP
PGCN (2 bytes)
reserved (1 byte)
PGN (1 byte)

FIG. 12

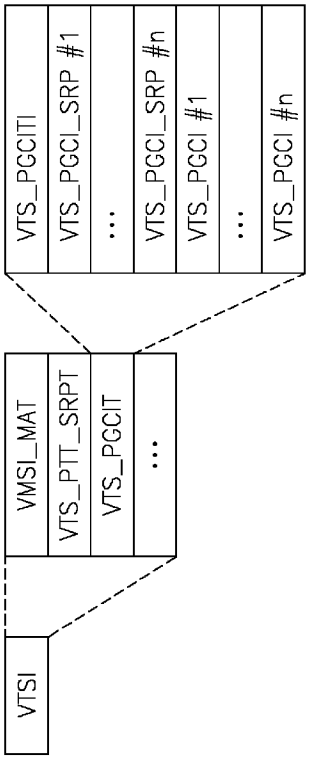

VTSI
VMSI_MAT
VTS_PTT_SRPT
VTS_PGCIT
...
VTS_PGCITI
VTS_PGCI_SRP #1
...
VTS_PGCI_SRP #n
VTS_PGCI #1
...
VTS_PGCI #n

FIG. 13

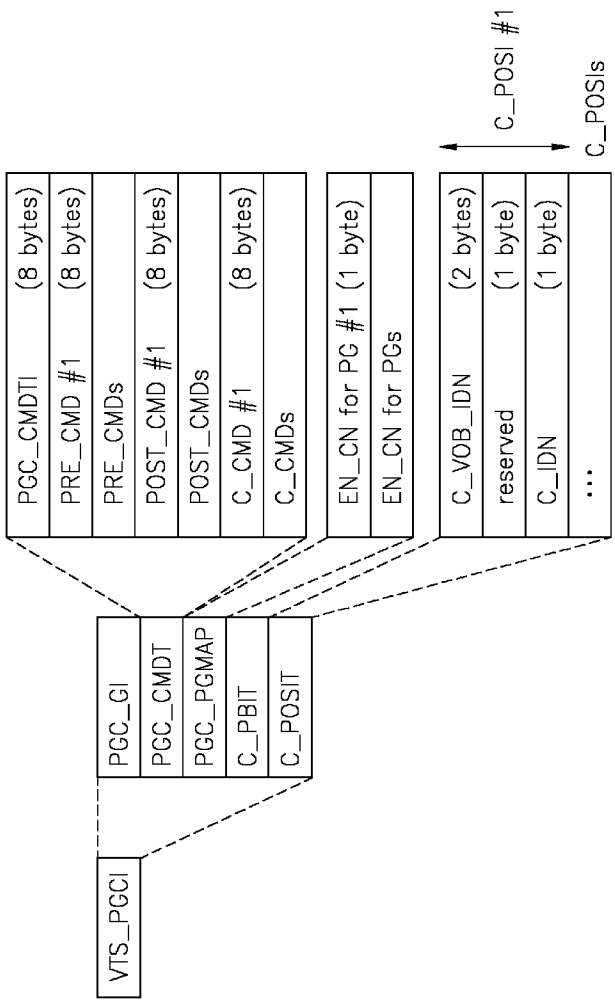

VTS_PGCI
PGC_GI
PGC_CMDT
PGC_PGMAP
C_PBIT
C_POSIT
PGC_CMDTI (8 bytes)
PRE_CMD #1 (8 bytes)
PRE_CMDs
POST_CMD #1 (8 bytes)
POST_CMDs
C_CMD #1 (8 bytes)
C_CMDs
EN_CN for PG #1 (1 byte)
EN_CN for PGs
C_VOB_IDN (2 bytes)
reserved (1 byte)
C_IDN (1 byte)
...
C_POSI #1
C_POSIs

FIG. 14

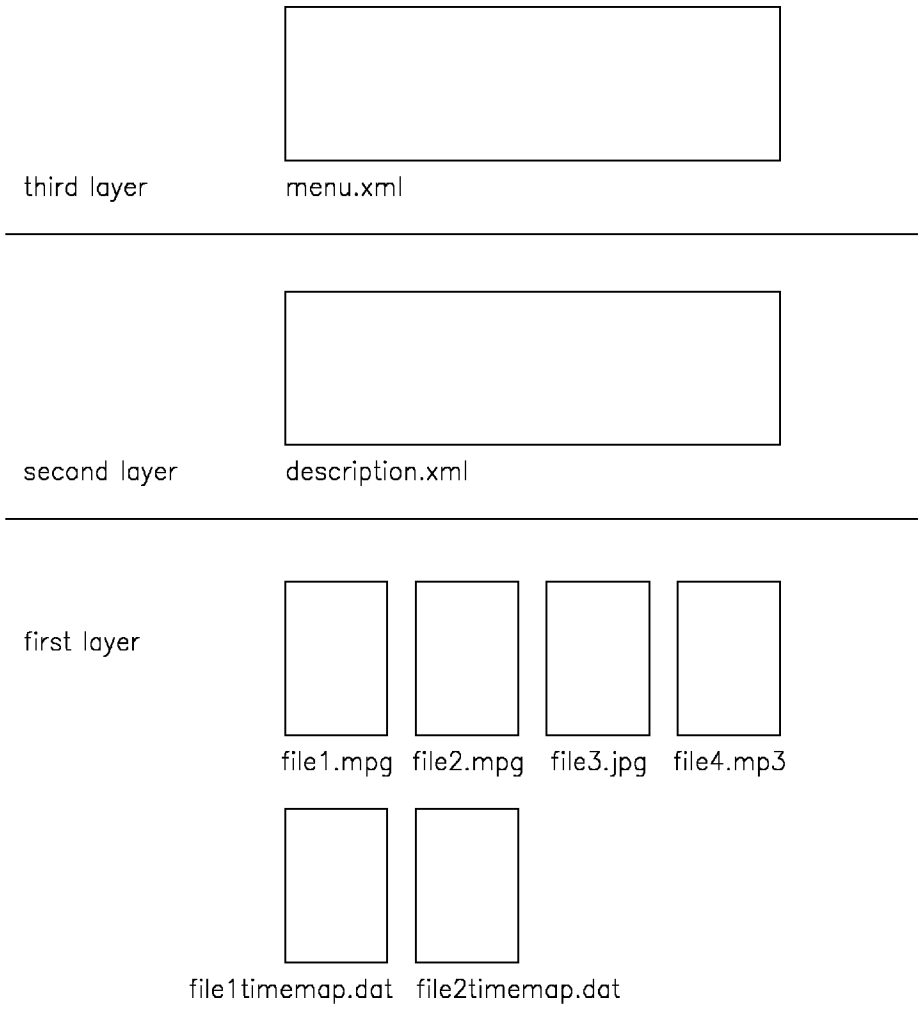

third layer
menu.xml
second layer
description.xml
first layer
file1.mpg
file2.mpg
file3.jpg
file4.mp3
file1timemap.dat
file2timemap.dat

FIG. 15

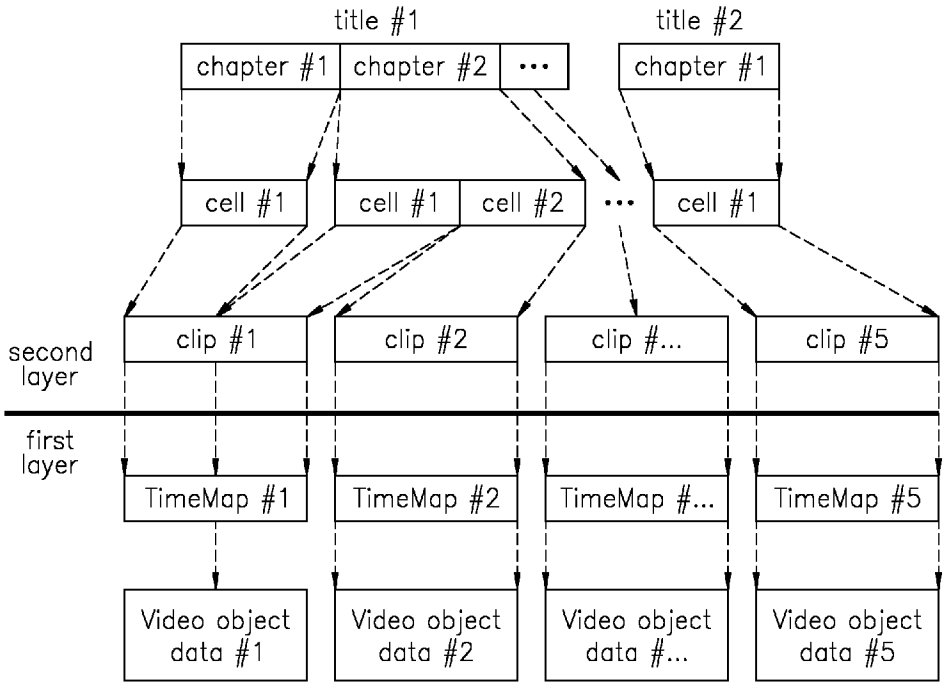

title #1
title #2
chapter #1
chapter #2
...
chapter #1
cell #1
cell #1
cell #2
...
cell #1
second layer
clip #1
clip #2
clip #...
clip #5
first layer
TimeMap #1
TimeMap #2
TimeMap #...
TimeMap #5
Video object data #1
Video object data #2
Video object data #...
Video object data #5

FIG. 16

TimeMap Entry number = m
Time Gap = n sec
Time Gap x 1 data position
Time Gap x 2 data position
Time Gap x 3 data position
Time Gap x 4 data position
Time Gap x 5 data position
...
Time Gap x m data position

FIG. 17

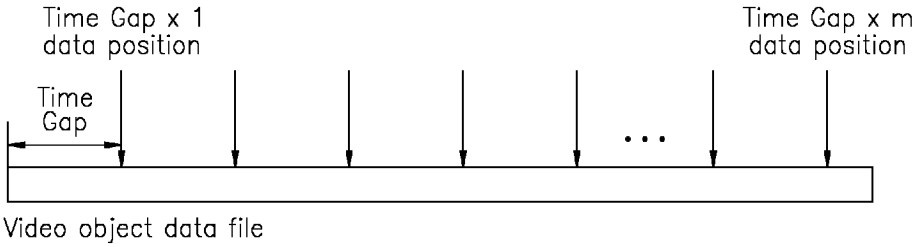
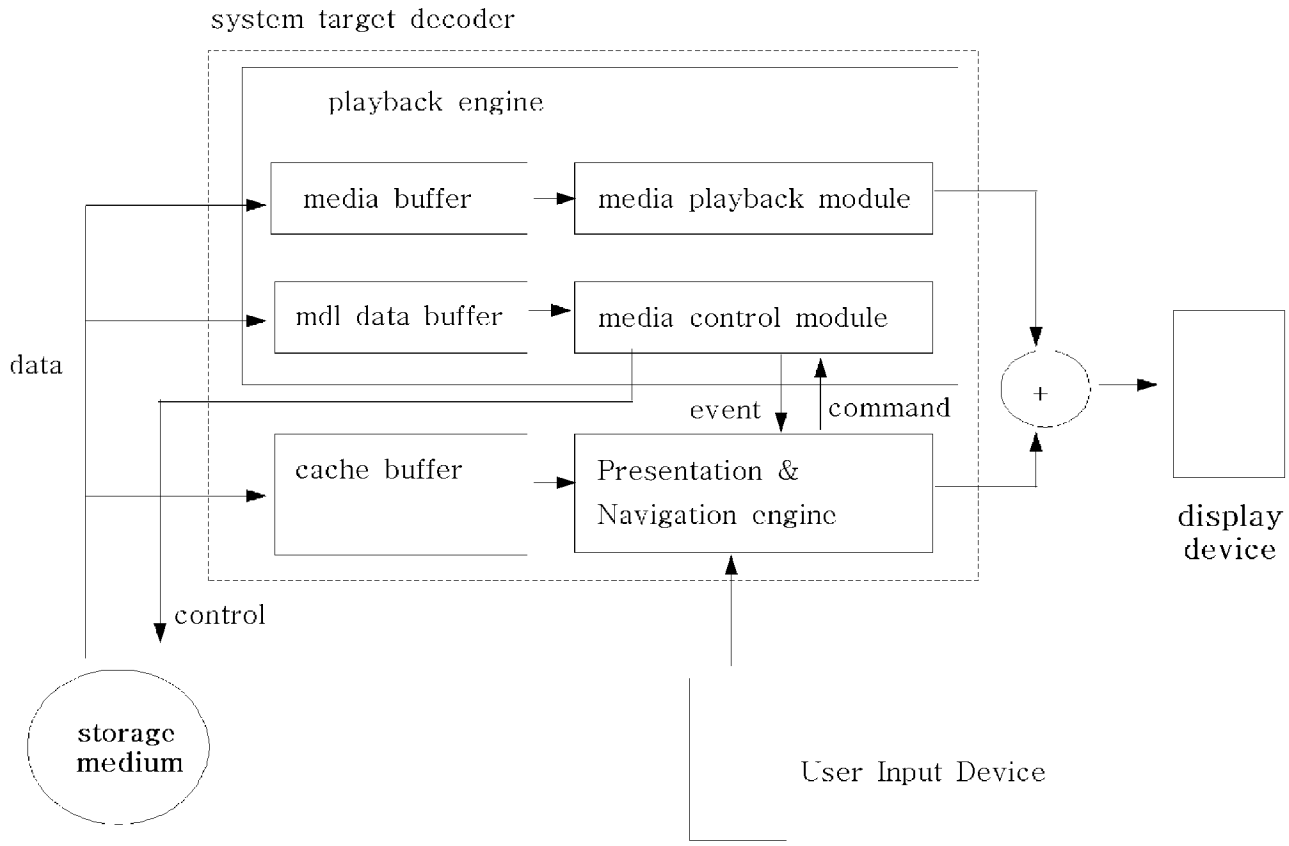


FIG. 18


system target decoder
playback engine
media buffer
media playback module
mdl data buffer
media control module
data
event
command
cache buffer
Presentation &
Navigation engine
+
display
device
control
storage
medium
User Input Device

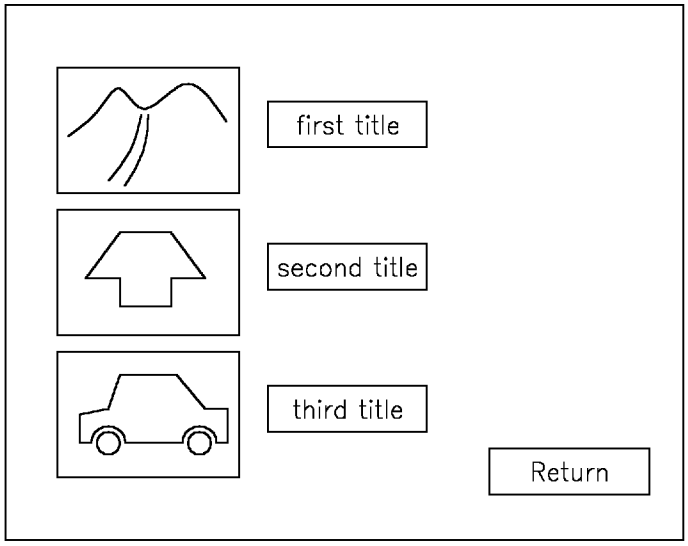



FIG. 19